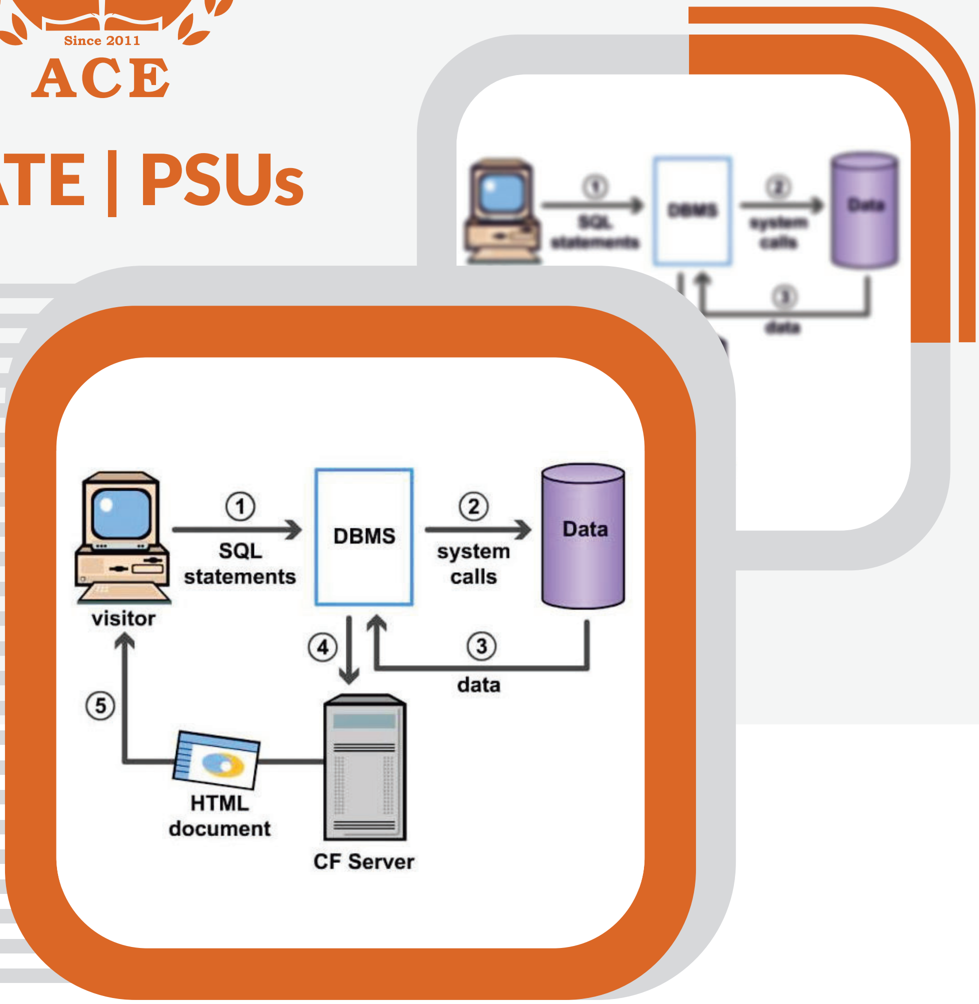
ACE ENGINEERING PUBLICATIONS
Since 2011
ACE

GATE | PSUs

COMPUTER SCIENCE & INFORMATION TECHNOLOGY



COMPUTER SCIENCE & INFORMATION TECHNOLOGY

DATABASE MANAGEMENT SYSTEMS

Volume-1 : Study Material with Classroom Practice Questions

Database Management Systems

(Classroom Practice Booklet Solutions)

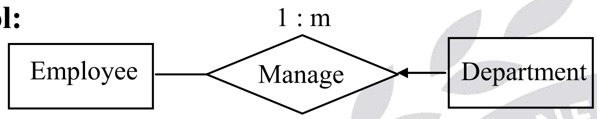
2. ER and Relational Model

01. Ans: (b)

Sol: Derived attribute is an attribute that derives its value from one or more attributes.

02. Ans: (b)

Sol:



03. Ans: (a)

Sol: As every specialized entity is subset of generalized entity, then the deletion of generalized entity requires the deletion of specialized entity.

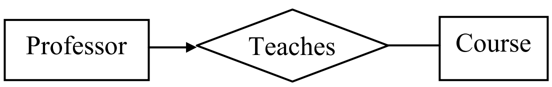
04. Ans: (c)

Sol:

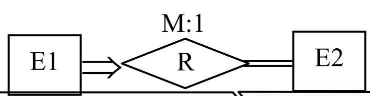
- Composite attribute is an attribute which is composed of other attributes.
- Multi valued attribute represented with double ellipse.
- Derived attributes represented with dotted ellipse

05. Ans: (b)

Sol:

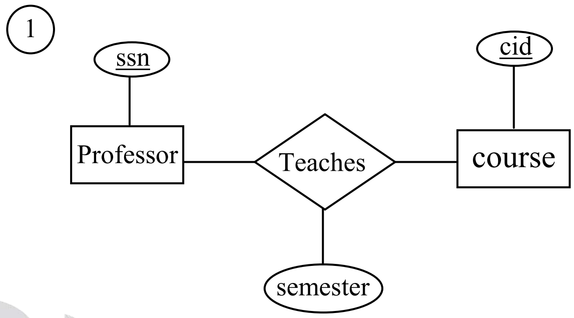


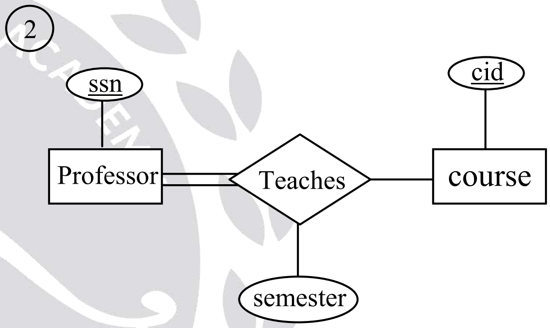
06. Ans: (a)

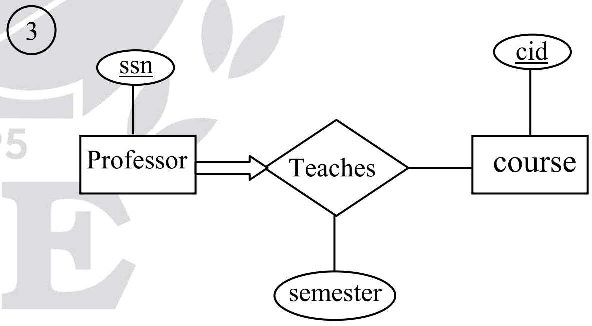


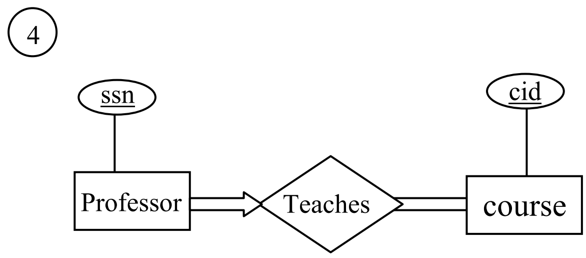
07. Ans:

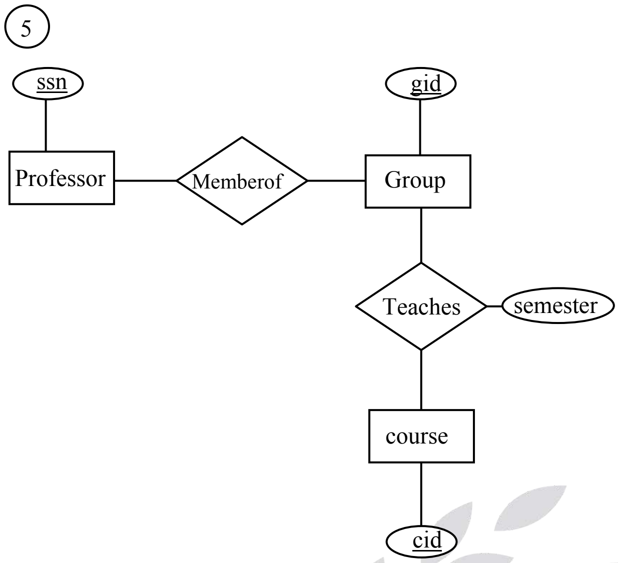
Sol:











08. Ans: (c)

Sol: Set of attributes which contains a candidate key is said to be a super key.

09. Ans: (b)

Sol: All the values present in Foreign key must present in primary key of the referenced relation.

10. Ans: (c)

Sol: It violates referential integrity constraint as it is updating in foreign key but not in primary key.

11. Ans: (c)

Sol: When parent is update, it requires child table to be updated simultaneously

12. Ans: (c)

Sol: 1. On removal of row (2,4), row (5,2) and (7,2) must also be deleted as they depend on value

2. On removal of row (5,2), row (9,5) must also be deleted as it depends on value 5.

13. Ans: 0

Sol: When <3, 8> is deleted, its related tuples in T_2 is (8, 3) and 3 is to set null. Hence the number of additional tuples to delete is 0

14. Ans: (a)

Sol: As the key constraint from professor, the maximum number of tuples possible in Teaches is number of tuples in professor.

15. Ans: (a)

Sol:

16. Ans: (a)

Sol: As C, A and B in total participation with R_1 and R_3 and there is key constraint, all these are represented with one relation and R_2 is re presented separately one relation.

17. Ans: (b)

Sol: There are some tuples of course may not participate with any tuple of professor, then cid is a key for the relation.

18. Ans: (a)

Sol: (AR_1B) will be one table as there is total participation and key constraint.

(CR_2) will be the second table as there is a key constraint.

19. Ans: (b)

Sol: As we get key and participation constraint from course to registration, therefore the number of tuples in registration will be equal to the tuples in the course table.

20. Ans: (b)

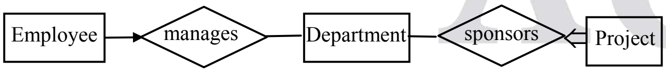
Sol: Strong entities E_1 and E_2 are represented as separate tables, in addition to that many to many relationship (R_2) must be converted as separate table by having primary key of E_1 and E_2 as foreign key. One to many relationship must be transferred to 'many' side table by having primary key of one side as foreign key. Hence we will have minimum of 3 tables.

21. Ans: (b)

Sol: Strong entities E1 and E2 are converted as separate tables. Since A23 is a multi valued attribute it should also be converted as separate table. Relationship R is transferred to 'm' side (E2).

22. Ans: 3

Sol: E-R model is

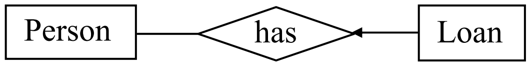


The minimum number of relations in relational model is 3.

1. (Employee, manages)
2. Department
3. (Project, sponsors)

23. Ans: (c)

Sol:



As there is a key constraint from Loan, Relationship can be merged with Loan entity set.

24. Ans: (b)

Sol: M, P are strong entities hence they must be represented by separate tables. M table is modified to include primary key of P side (i.e P1). N is weak entity, and it is modified to include primary key of P (i.e P1).

25. Ans: (a)

Sol: M and P are strong entities hence they must be represented as separate tables. To include R1, M table is modified to accommodate primary key of P side (i.e P1) as foreign key. N is weak entity, so modify N to accommodate primary key of P (i.e P1) as foreign key.

Therefore tables are (M1, M2, M3, P1), (P1, P2), (N1, N2, P1). So correct answer is (M1, M2, M3, P1).

26. Ans: (b)

Sol:

- An Entity type is represented with one relation.
- Key attribute becomes primary key for the relation
- Composite attribute is represented with set of simple attributes
- Weak entity is represented always as a child table

3. Functional Dependencies

01. Ans: (d)

Sol: As 'BC' is key $BC \rightarrow A$ is satisfied

02. Ans: (b)

Sol: Based on the table values given in query and guidelines below, answer is b. YZ is having unique combination, and Y is also having unique values. Hence $YZ \rightarrow X$, $Y \rightarrow Z$ are possible.

03. Ans: (b)

Sol: A functional dependency $X \rightarrow Y$ is satisfied if any two tuples agree on X then they must also agree on Y.

04. Ans: (c)

Sol: AF^+ = AFDE not ACDEFG as given.

05. Ans: (c)

Sol: A functional dependency $X \rightarrow Y$ is said to be trivial iff $Y \subseteq X$.

06. Ans: (b)

Sol: CD^+ from functional dependencies (FDs) = **C**DE**A**B, it includes RHS attributes AC so it can be derived from FDs BD^+ from functional dependencies (FDs) = BD only, RHS attributes CD are not included in the closure hence it cannot be derived BC^+ from functional dependencies (FDs) = BCD EA, it includes RHS attributes CD, so it can be derived from FDs AC^+ from functional dependencies (FDs) = ACBDE, it includes RHS attributes BC so it can be derived from FDs

07. Ans: (c)

Sol: AC^+ contains I then $AC \rightarrow I$ dependency is possible.

08. Ans: 24

Sol: $2^4 + 2^4 - 2^3 = 24$.

09. Ans: (b)

Sol: As 'K' is independent attribute, key is ABDK.

10. Ans: (d)

Sol: ABD^+ = A, B, C, D, E.

11. Ans: (b)

Sol: $ACEH^+$ contains all the attributes of R.

12. Ans: (d)

Sol: Closure of $AEH^+ = BEH^+ = DEH^+$ = A, B, C, D, E, H. If any closure includes all attributes of a table then it can become candidate key of the table. Closure of AEH, BEH, DEH includes all attributes of table. Hence they are candidate keys.

13. Ans: (b)

Sol: Set of attributes which contains a candidate key is called super key.

14. Ans: 3

Sol: The candidate keys are

F

AB

CB

15.

Sol: CK: ACD, BCD, ECD.

16. Ans: 6

Sol: AB, AD, EB, ED, CB, CD.

17. Ans: 2

Sol: D, AH

18. Ans: (b)

Sol: The number of super keys are A, B, C, AB, AC, BC, ABC.

19. Ans: (a)

Sol: $D \to E$ of F is not covered by G.

20. Ans: (c)

Sol: $D \to C$ in set2 and $C \to D$ in set1 not covered by each other.

21. Ans: (d)

Sol: $AB \to C$, $A \to BC$ both can be determined from remaining set of FD's.

22. Ans: 5

Sol: $AC \to D$ can be eliminated, it can be derived from $A \to B$ and $CB \to D$ using augmentation and transitive rule.

$A \to B \Rightarrow AC \to BC$

$\Rightarrow AC \to D$

And remaining FD's are not possible to eliminate

$\therefore$ 5 FD's are there in minimal cover.

23. Ans: (c)

Sol: The minimal set of F is $A \to B$, $B \to C$ and $AB \to C$ is redundant.

24. Ans: (a)

Sol: As $V \to W$, delete W from $VW \to X$ results in $V \to X$

As $V \to X$, delete X from $Y \to VX$ results in $Y \to V$

The irreducible set is

$V \to W$

$V \to X$

$Y \to V$

$Y \to Z$

25.

Sol:

$$\left.\begin{array}{l} AD \to CF \\ C \to B \\ B \to E \end{array}\right\} \text{Canonical set}$$

26.

Sol:

$$\left.\begin{array}{l} A \to BC \\ AE \to H \\ C \to D \\ D \to G \\ E \to F \end{array}\right\} \text{Minimal set}$$

4. Normalization

01.

Sol: 1. C.K = BD, Lossy, Dependency preserving

2. C.K = AB, CB, Loss-less,
   Not Dependency preserving

3. C.K = A, C, Loss-less,
   Dependency preserving

4. C.K = A, Loss-less,
   Not Dependency preserving

5. C.K = A, Lossy,
   Not Dependency preserving

02. Ex: F: {AB→C, A→D}

03. Ex: F: {AB→C, C→D}

04.

Sol: R is in 1NF ∴ decompose to 2NF

A^+ = {A, D, E, I, J} R_1 = 2NF

B^+ = {B, F, G, H} R_2 = 2NF

R_1 (A D E I J)
- D^+ R_4 (D I J)
- R_5 (A E D)

R_2 (B F G H)
- F^+ R_6 (F G H)
- R_7 (B F)

{A, B, C} R_3 BCNF

Then decompose into 2NF

R_1 (ADEIJJ)
R_2 (BFGH)
R_3 (ABC)
3NF also in BCNF
R_3 (ABC)
R_4 (DIJ)
R_5 (AED)
R_6 (FGH)
R_7 (BF)
AB^+ is key.

05.

Sol: Candidate key: AC

A^+ = (ABE) R_1, C^+ = (CD) R_2

(ACF) R_3

06.

Sol: (1) C → D
C → A
B → C
C.K: B, 2NF but not 3NF

(2) 2NF but not 3NF as no partial dependency CK: BD.

(3) R is in 3NF but not in BCNF

(4) C.K = A

(5) Candidate Keys = AB, CD, BC, AD
R is in 3NF but not in BCNF.

07. Ans: (d)

Sol: 3NF requires for a non trivial FD of the form X →A then X is super key (or) A is prime attribute.

08. Ans : (a)

Sol: As given client id and order id together is a key and it is possible to determine Firstname, Lastname of a client using his client id, then we have the dependency clientid → Firstname, Lastname which is a partial functional dependency. Hence the relation is in 1 NF.

09. Ans: (b)

Sol: (Volume, Number) $\rightarrow$ Year is a partial functional dependency. So, the given relation is in 1 NF but not in 2 NF.

10. Ans: (c)

Sol: R is in 1NF as A→FC and B→E are partial dependencies

11. Ans: (d)

Sol: Relation R_1 satisfies A→B, B→C and C→AB dependencies and all the determinants are super keys. Hence the relation is in BCNF.

12. Ans: (c)

Sol: In CD→B; B is prime attribute.

13. Ans: (b)

Sol: rollno, courseid is superkey in rollno, courseid→email, rollno is prime attribute in email→rollno.

5. Structured Query Language (SQL)

01. Ans: (b)

Sol: The result of the query is

A	B	C
1	4	4
2	5	1
2	5	3
3	5	2

02. Ans: (c)

$$\text{avg (marks)} = \frac{(10+0+30+0)}{4} = 10$$

03. Ans: (b)

Sol: Select clause contains either aggregate function or the attributes that appear in group by clause.

04. Ans: (c)

Sol: sum (rating)/count(0) is smaller value than avg(rating).

05. Ans: (a)

Sol: If A→B does not hold, we expect some output.

06. Ans: (c)

Sol: Union operator eliminates the duplicates.

07. Ans: (c)

Sol: The result of the query is

e1. empno	e2.empno
326	321
350	323
351	323
351	350

08. 2

Sol: It returns two rows.

Student – Name	**Sum(P.Marks)**
Raj	4
Rohit	2

09. Ans: (c)

Sol:

R_1

A	B
1	5
3	7

R_2

A	C
1	7
4	9

$R_1 \bowtie R_2$

A	B	C
1	5	7
3	7	null
4	null	9

10. Ans: 8

Sol: Full outer join of R and S will give T relation. Here NULL entries are taken for R and S to include all missing instances of common attribute A while joining R and S.

11. Ans: (d)

Sol: Inner query retrieves average salary of all employees. Outer query computes average salary of all male employees. Therefore it retrieves department names where average salary of male employees is more than the average salary in the company.

12. Ans: (b)

Sol: The condition is B>any (1,2,1,3,2,4) and the output will be 4.

13. Ans: (d)

Sol: Finds rating and average age of each rating of those sailors whose age is >= 18 and there is atleast two sailors for each rating.

The output is

S.rating	**avg(S.age)**
7	40
8	40
3	45

14. Ans: (a)

Sol: all(empty) returns true always.

15. Ans: (c)

Sol: Inner Query finds the number of guests trained by JULIO

16. Ans: 7

Sol: The output of the query is

<u>ta.player</u>
Klose
Ronaldo
G muller
Fontaine
Pele
Klismann
Kocsis

17. Ans: (d)

Sol:

S_1

Name	Age	Rank
A	16	4
B	17	3
C	15	1
D	21	6
E	22	7

S_2

Name	Age	Rank
A	16	4
B	17	3
C	15	1
D	21	6
E	22	7

When the query executes on the above table of data returns A,D,E in the output, those are students with higher rank than all students with age <18.

Exercise - 01

01.

Sol: Display the details of all hotels, located in London

02.

Sol: Display name & address of all guests in ascending order of their name living in London.

03.

Sol: Display the details of all rooms of price below 40 & type of the room is either 'D' or 'F' in ascending order of their prices.

04.

Sol: Display all booking details whose check out time not specified.

05.

Sol: Finds total price of all rooms of type 'D'.

06.

Sol: Find the number of guests who have a booking on 1st august or in the month of august.

07.

Sol: List the type & price of all rooms in Grosvenor hotel.

08.

Sol: List all the guest details who have a booking for the current date in "Grosvenor Hotel'

Select * from guest where guestNo in ( ) &. hotelNo = ( )

= operator – when only 1 value is return.

09.

Sol: List the details of all rooms at the 'Grosvenor hotel' including the name of the guest staying in the room, if the room is occupied.

10.

Sol: Finds total price of all rooms in 'Grosvenor hotel' if the room is currently occupied.

11.

Sol: Find number of rooms in each hotel located in London.

12.

Sol: Find the maximum number of rooms booked in London of same type

A	100(Room Type)
B	200
C	150
D	75

Maximum 200. It displays total no. of bookings for the most popular room in city of London.

13.

Sol: Find total price of all rooms, that are not currently booked in each hotel.

6. Relational Algebra & Calculus

01. Ans: (b)

Sol: As list1 $\subseteq$ list2 the result will be equal to

$$\prod_{\langle list1 \rangle} R$$

02. Ans: (d)

Sol: Relational Algebra eliminate duplicates always.

03. Ans: (c)

Sol: If (X=Y) $\subseteq$ Z then the expressions to be true.

04. Ans: (c)

Sol: In the order of evaluation, first we perform selection, then we perform perfection.

05. Ans: (a)

Sol: $\Pi_B(r_1) - \Pi_C(r_2) = \phi$ is always true. Because 'B' is foreign key referencing 'C' , so 'C' must be a primary key, 'B' cannot have a value that is not available in 'C'. Hence operation $\Pi_B(r_1) - \Pi_C(r_2)$ is always ϕ.

06. Ans: (a)

Sol: Common column between tables 'R' and 'S' is attribute B. In table 'R' B is primary key (B→A, A→C). In table 'S' B is foreign key so join is performed on attribute B. Therefore maximum tuples possible in the output is equal to rows in Table S (as it has less number of rows, provided B values are not repeated in table 'S').

07. Ans: (b)

Sol: bal < 0 filter rows from account ∞ depositor from which we can operate on few rows to filter b city = "Agra".

08. Ans: (d)

Sol: Minus operator indicates rows available in LHS table but not in RHS table. In this expression, LHS table produces all female students, RHS table consists students with less marks hence it produces names of all girl students with more marks than all the boy students.

09. Ans: (a)

Sol: Apply first cross product then apply filter. Cross product yields 10 rows, then you play filter A.ID>40 or C.ID<15 produces 7 rows.

$A \cup B \Rightarrow$ A. Id

12, 15, 25, 98, 99

$|x|$ is cross product followed selection and projection.

A. Id	B.Id	X gives (cross product)		10 rows		Condition
			A. Id	B. Id	A. Id > 40 V	C.Id <15
	5 rows	2 rows		12	10	√
				12	99	×
				15	10	√
				15	99	×
				25	10	√
				25	99	×
				98	10	√
				98	99	√
				99	10	√
Result contains 7 rows				99	99	√

10. Ans: (a)

Sol: Division operator is used to compare a value with all the values of other relation.

First expression returns sid's of sailors who reserved all boats called Ganga, and its outer query returns those sailor names.

11. Ans: 4

Sol: The output of T_1 is: courseName

CA
CB
CC

the output of T_2 is: StudentName

SA
SC
SD
SF

12. Ans: (c)

Sol: P. duration = 3 months selects all projects of duration 3 months

T. pname = P.name selects project names in the output.

13. Ans: (a)

Sol: SQL, Relational algebra, tuple relational calculus and Domain relational calculus all is representing the same. i.e., all these expressions representing to find the distinct names of all students who score more than 90% in the course numbered 107.

14. Ans: (d)

Sol: Minus operator indicates, rows available in LHS table but not in RHS table. In this expression, LHS table produces all female students, RHS table consists students with less marks hence it produces names of all girl students with more marks than all the boy students.

15.

Sol: Result of given query

P	Q	U
2	b	5
2	b	6
2	c	5
3	d	6

16. Ans: 2

Sol: Relational calculus eliminate the duplicates.

$\{T / \exists B \in \text{Book}\ (T.\text{Title} = B.\text{Title})\}$

17.

Sol:

(a) $$\left\{\begin{array}{l} T \mid T \in \text{Employee} \wedge \exists M \in \text{Manger} \\ \left(\begin{array}{l} M.\text{Manager\_name} = \text{'Jones'} \wedge \\ T.\text{Person\_name} = M.\text{Person\_name} \end{array}\right) \end{array}\right\}$$

(b) $$\left\{\begin{array}{l} T \mid \exists\ E \in \text{Employee} \\ (T.\text{City} = E.\text{City}) \wedge \exists M \in \text{Manager} \left(\begin{array}{l} M.\text{manager\_name} \\ = \text{'Jones'} \wedge E.\text{person\_name} \\ = M.\text{person\_name} \end{array}\right) \end{array}\right\}$$

(c) $$\left\{ T \mid \exists\ E \in \text{Employee} \left(\begin{array}{l} M.\text{person\_name} \\ = \text{'Jones'} \wedge T.\text{Manager\_name} \\ = E.\text{Manager\_name} \end{array}\right) \right\}$$

7. Transactions & Concurrency Control

01. Ans: (d)

Sol: Transaction T_3 perform read on A, which is updated by T_1 and committed before T_1 does.

02. Ans: (a)

Sol: As $R_3(x)$ is dirty operations which read $W_1(x)$ and is committed before T_1. Hence schedule is non-recoverable.

03. Ans: (c)

Sol: A recoverable schedule is one where for each pair of transactions T_i and T_j such that T_j reads a data item previously written by T_i, the commit operation of T_i appear before the read operation of T_j.

04. Ans: (c)

Sol: A schedule is said to be strict if a value written by a transaction T is to be read or written by another transaction until either T commits or aborts.

05. Ans: (d)

Sol: A recoverable schedule is one where for each pair of transactions T_i and T_j such that T_j reads a data item previously written by T_i, the commit operation of T_i appear before the read operation of T_j.

A schedule is said to be strict if a value written by a transaction T is to be read or written by another transaction until either T commits or aborts.

06. Ans: (b)

Sol: The number of serial schedules are 2

The number of concurrent schedules are $= \frac{(5+3)!}{5!*3!} = 56$

Then, the total number of non serial schedules are = (number of concurrent schedules – number of serial schedules)

= 56 – 2 = 54

07. Ans: (b)

Sol: Every cascadeless schedule is recoverable but need not vice versa.

08. Ans: (a)

Sol: R1(X), R1(Y), R3(Y), W2(Y), W1(X), W3(X), R2(X), W2(X).

R3(Y), W2(Y) are conflicting hence serializability says **T3 must be before T2** in any serial schedule.

W1(X), W3(X) are conflicting hence serializability says **T1 must be before T3** in any serial schedule.

W3(X), R2(X) are conflicting hence serializability says **T3 must be before T2** in any serial schedule.

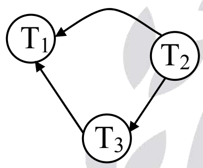
The serial schedule T1, T3, T2 is satisfying all the above conditions. Directed graph for the non-serial schedule will also give same sequence.

09. Ans: (a)

Sol: S1:

T_1	T_2	T_3
R(x)		
		R(y)
		R(x)
	R(y)	
	R(z)	
		W(y)
	W(z)	
R(z)		
W(x)		
W(z)		

Precedence graph

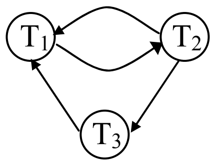


S1 is conflict serializable to $T_2 \rightarrow T_3 \rightarrow T_1$

S2:

T_1	T_2	T_3
R(x)		
		R(y)
	R(y)	
		R(x)
R(z)		
	R(z)	
		W(y)
W(x)		
	W(z)	
W(z)		

Precedence graph



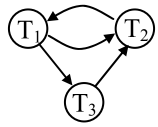
S2 is not conflict serializable

10. Ans: (d)

Sol:

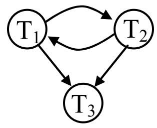
Option (A)

T_1	T_2	T_3
R(x)		
	R(x)	
W(x)		
		R(x)
	W(x)	



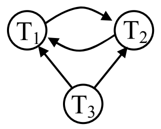
Option (B)

T_1	T_2	T_3
	R(x)	
R(x)		
	W(x)	
		R(x)
W(x)		



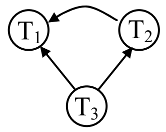
Option (C)

T_1	T_2	T_3
		R(x)
	R(x)	
R(x)		
	W(x)	
W(x)		



Option (D)

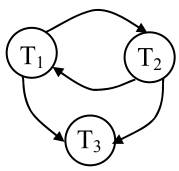
T_1	T_2	T_3
	R(x)	
	W(x)	
R(x)		R(x)



Conflict Serializable

11. Ans: (c)

Sol: Precedence graph is



12. Ans: (d)

Sol: S1 and S2 are conflict equivalent to serial schedule T2, T3, T1.
S3 is not conflict equivalent as 2RA, 3WA (T2<T3) and 3WA, 2WA (T3<T2) are the conflict operations. There is no serial schedule that satisfies both T2<T3 and T3<T2.

13. Ans: (b)

Sol: As there is a cycle in precedence graph, the schedule is not conflict serializable, but satisfying view rules.

14.

Sol: (a) Not Conflict Serializable,
Not View Serializable,
Recoverable, Avoids Cascading aborts,
Not strict.

(b) Not Conflict Serializable,
Not View Serializable,
Not strict,
Recoverable, cascading aborts

(c) Conflict Serializable,
View serializable,
Serializable,
Recoverable,
Avoids cascading aborts,
Not strict

(d) Not Conflict Serializable,
Views serializable through Thomas write rule, Serializable,
Recoverable,
Avoids cascading aborts,
Not strict

(e) Conflict Serializable,
View Serializable,
Serializable,
Recoverable,
Avoids cascading aborts,
strict

T_1	T_2	T_3
R(A)		
	W(A)	
		R(A)
W(A)		
		W(A)

(f) Conflict Serializable,
View serializable,
Serializable,
Recoverable,
No need cascading aborts,
Strict

(g) Not Conflict Serializable,
Not View Serializable,
Not Serializable,
Recoverable,
No need cascading aborts,
strict

(i) Conflict Serializable,
View Serializable,
Serializable,
Not Recoverable,
Not Avoid cascading aborts,
Not Strict

15. Ans: (c)

16. Ans: (c)

Sol: An older transaction requesting a data item held by an younger T_x need to wait.

17. Ans: (b)

Sol: Construct the wait-for-graph, contains edges from T1→T2, T2→T3, and T4→T2 and there is no circular wait, hence no deadlock.

18. Ans: (d)

Sol: W1(a), R2(a) and W2(b), R1(b) are conflicting, going by the principles of serializability it is not serializable and cannot occur in 2PL

8. Indexing

01. Ans: 400

Sol: Blocking factor = 512/20=25

Number of data blocks = 10000/25=400

In primary index, the number of index records= 400 which is number of blocks in multi level index.

02. Ans: (c)

Sol: Since block size is 2^{10} (=1024 bytes), $2^m \times 2^n = 2^{10}$. Only option (c) satisfies with m = 8 and n = 2.

03. Ans: 50

Sol: Order of non-leaf node is

$(n\times 8) + (n-1)\ 12 \le 1024$

$8n + 12n - 12 \le 1024$

$20n \le 1036$

$n \le 51$

maximum number of keys possible is :50

04. Ans: 52

Sol: Key =8 Block size = 512,

Block pointer = 2 bytes, the order of B^+ tree is maximum number of block pointers in it. (Let 'n')

$n * 2 + (n-1)\ 8 \le 512$

$2n + 8n - 8 \le 512$

$10n \le 520$

$n \le 52$

05. Ans: (c)

Sol: $n*5+(n-1)*(10+8) \le 512$

$5n+16n-18 \le 512$

$23n \le 530$

$n \le 23.$

06. Ans: (b)

Sol: n × P + (n-1)k <=B. Where n is order of the tree, P is block pointer, k is key value and B is block size.

Therefore n × 6 + (n – 1)9 <=1024.

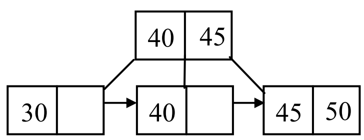
n = 1033/16 = 64 (approximately)

07. Ans: 5

Sol: The nodes to access all records with a "search key greater than or equal to 7 and less than 15" is (9), (5), (5, 7) (9, 11) and (13, 15).

08. Ans: (b)

Sol: The resultant tree after the insertion is

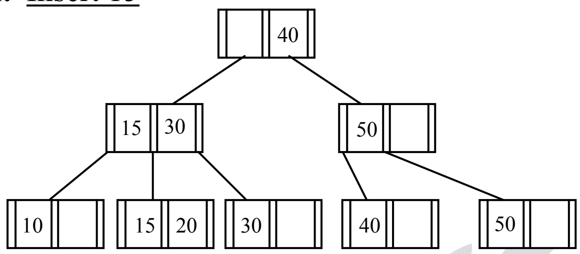


09. Ans: (d)

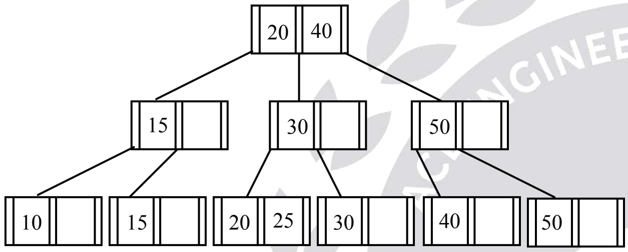
Sol: Deleting ‘10’ from internal node requires 10 to be replaced with copy of 13.

10. Ans: (a)

Sol: Insert 15



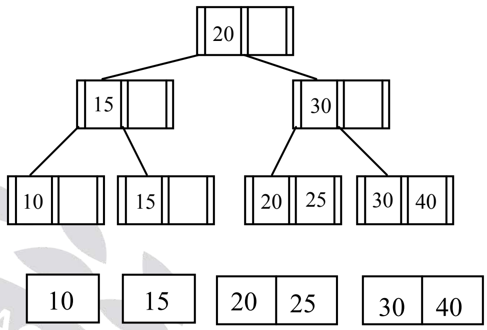
Insert 25



11. Ans: (a)

Sol: If we remove K50 in index node, there is no change in height as still root, index and leaf nodes exist.

Leaf nodes are now, after Delete 50, the B^+ is:



Hence Root now consists: 20

(i) is true, (ii) is true but (iii) is not true